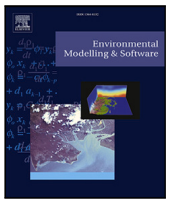
Environmental Modelling and Software 185 (2025) 106288

Contents lists available at ScienceDirect

Environmental Modelling and Software

journal homepage: www.elsevier.com/locate/envsoft



Solving the Master Equation on river networks: A computer algebra approach

Samuele De Bartolo [a,*], Gaetano Napoli [b], Stefano Rizzello [a], Raffaele Vitolo [c,d]

[a] Dipartimento di Ingegneria per l'Innovazione, EUropean Maritime Environmental Research (EUMER), Università del Salento, Via per Monteroni s.n.c., Lecce, 73100, Italy
[b] Dipartimento di Matematica e Applicazioni "Renato Caccioppoli", Università degli Studi di Napoli "Federico II", Complesso Monte S. Angelo, Napoli, 80126, Italy
[c] Dipartimento di Matematica e Fisica "E. De Giorgi", Università del Salento, Via per Arnesano s.n.c., Lecce, 73100, Italy
[d] Istituto Nazionale di Fisica Nucleare, Sezione di Lecce, Via per Arnesano s.n.c., Lecce, 73100, Italy

ARTICLE INFO

Dataset link: https://github.com/stefano-rizzello/MasterEqSoftware

Keywords:
Surface Hydrology
Fluvial Hydraulics and sediment transport
High-performance computing
River networks and Landscape analysis
Computer algebra
Reduce
Mathematica
Diffusion models
Master Equation

ABSTRACT

We describe the algorithms and the software that have been used in a new computational method based on the use of Master Equations. Our computer algebra procedures simulate the diffusion of a pollutant in river networks. The representation of river networks as trees makes the derivation of governing equations for pollutant transport an easy task. This includes mass balance equations that account for the sources, sinks, and transport of pollutants in the river network. In two previous papers we described the model and some simulations obtained from our software. In this paper we describe two software libraries, respectively for the Reduce and the Mathematica computer algebra systems, that have been developed on the basis of our model. The libraries can be found in our `GitHub` repository.

1. Introduction

River networks are the support through which rainwater and other waters are distributed spatially and temporally on ground surface. The transport processes in this complex system of networks, represented by a branched series of arcs (streams or channels) and node (junctions), are strongly linked to the residence times of the flow volumes and all the additional interacting masses, including pollutants.

River networks, in their branched structure, present topological characteristics, which can be hierarchized through various sorting systems, primarily the Horton–Strahler system. Such systems, although efficient from a quantitative point of view, still show uncertainties in the area of river geomorphological scaling information (De Bartolo et al., 2009). Currently, these uncertainties are being overcome through the use of increasingly sophisticated spatial information systems, capable of providing details even at metric scales (LiDAR) (De Bartolo et al., 2016; Costabile et al., 2019, 2024).

In this context, complex river systems are dynamic environments susceptible to pollution from various sources. Therefore, it is quite challenging to analyze complex phenomena such as pollutant transport on scales ranging from channel to basin network (Rizzoli and Young, 1997; Harrison et al., 2023; Villa et al., 2009; Argent et al., 2009). The most common ones are chemical pollutants, which often originate from industrial discharges, agricultural runoff, urban storm-water and improper waste disposal as ammonia, nitrogen and phosphorus compounds (Chatwin and Allen, 1985; Duarte and Boaventura, 2008; Deng and Jung, 2009; Mazaheri et al., 2015). Thus, analyses and estimates of the damages to the water quality due to pollution could play an essential role in establishing governmental regulations for environmental protection (Kachiashvili et al., 2007; Chatwin and Allen, 1985; Duarte and Boaventura, 2008; Barati Moghaddam et al., 2017).

Estimating pollution using physical models in rivers involves fundamental concepts of fluid mechanics, mass transport and chemical interactions to simulate and predict the behavior and dispersion of contaminants within the river system (Alley, 2007; Altenburger et al., 2015; Novotny, 1994; Olowe and Kumarasamy, 2021; Ani et al., 2009; Rizzoli and Young, 1997; Harrison et al., 2023; Villa et al., 2009; Argent et al., 2009; Badrzadeh et al., 2022).

On river networks, it is important to consider the scale at which the hydraulics and hydrological processes are defined. These complex systems are analyzed at two main scales: the channel scale (Rodriguez-Iturbe and Rinaldo, 2001) and the basin scale (Rodriguez-Iturbe et al., 2009; Rinaldo et al., 2018).

* Corresponding author.
E-mail addresses: samuele.debartolo@unisalento.it (S. De Bartolo), gaetano.napoli@unina.it (G. Napoli), stefano.rizzello@unisalento.it (S. Rizzello), raffaele.vitolo@unisalento.it (R. Vitolo).

https://doi.org/10.1016/j.envsoft.2024.106288
Received 30 July 2024; Received in revised form 7 November 2024; Accepted 3 December 2024
Available online 17 December 2024

At the channel scale, the dispersion is primarily related to physical phenomena of motion resistance (Benedini and Tsakiris, 2013; Rodriguez-Iturbe et al., 2009). Specifically, it is closely linked to solid transport and equilibrium conditions of the longitudinal and transversal channel geomorphology (Rodriguez-Iturbe et al., 2009; Rinaldo et al., 2018; Benedini and Tsakiris, 2013; Rizo-Decelis et al., 2017). Regarding the basin scale, numerous applications in the fluvial environmental domain (Botter et al., 2011, 2010; Rodriguez-Iturbe et al., 2009; Rinaldo et al., 2018) show that the pollutant dispersion, is influenced by the number of tributaries at junctions (Rodriguez-Iturbe et al., 2009; Rinaldo et al., 2018; Rodriguez-Iturbe and Rinaldo, 2001) and generally by the river geomorphology as well as by the condition of motion (Botter et al., 2011, 2010; Mazaheri et al., 2015).

In view of the complexity of the phenomena, some Authors chose to simplify the models using the concept of Master Equation. These discrete equations are used to describe the temporal evolution of a dynamical system as a probabilistic combination of states at any given time (Botter et al., 2011, 2010).

The Master Equation is a discrete version of the advection equation; hence, advection is the principal physical phenomenon on which the Master Equation is based. In this way, conservation of mass and momentum are approximately preserved in the discrete model (Botter et al., 2011, 2010).

The factors that could influence the transport of pollutants in a river, like river geometry, diffusion coefficients, flow velocity and others, can be kept into account by increasing the number of nodes and/or varying the transition probability between nodes.

In our previous papers (Rizzello et al., 2023a,b) we used the Master Equation at channel and basin scales in order to model river networks. Our main tool was provided by two software packages, written in Mathematica and in Reduce (Reduce project page, 2023), that allowed us to calculate and solve the Master Equation in simulated environments and in real fluvial channel networks.

We are aware of the fact that the above approach might seem to be a simplistic model of a complex physical reality; but the comparison with experimental data supports the fact that the model is enough to reproduce physical pollution phenomena. On the other hand, the use of Master Equation in modeling river networks dates back to Botter et al. (2011, 2010), where it was shown that there is a good correspondence between observed data and the outcome of simulations; such evidences were later confirmed also by us (Rizzello et al., 2023a,b).

The main purpose of the present paper is that of a complete description of the computer algorithms and software that we developed for an implementation of the generation of the Master Equation and its solution.

In essence, we implemented the river network as a binary tree, where to every pair of nodes it is associated a transition probability. Given an initial pollutant input, our algorithm is recursively computing the evolution of the pollutant through the nodes at each consecutive time step. To this aim, computer algebra systems turn out to provide quite efficient development environments. More precisely, we programmed two implementations, in Mathematica and in Reduce (Reduce project page, 2023), and we are making them available in a `GitHub` repository (Rizzello et al., 2024) in order to make our results reproducible by any interested scientist and to give the possibility of future enhancements of our results.

In previous research works (Rizzello et al., 2023a,b) we presented several simulations; here, we consider one of them and we describe the Mathematica and Reduce programs by which we did the simulation. The programs use our packages in order to calculate and solve the Master Equation for the simulated river network.

2. Methods

Master Equations represent a foundational concept in various scientific disciplines, from physics and chemistry, to biology and economics and various branches of mathematics (Botter et al., 2011, 2010; Zaramella et al., 2015). The Master Equation has been extensively used in various fields to describe the time evolution of probability distributions of the physical quantities that are involved. Its application to environmental systems, particularly pollutant advection/diffusion in rivers, is relatively recent (Botter et al., 2011, 2010; Kadanoff and Swift, 1968; Cárdenas et al., 2023). Previous studies have employed simpler models, such as the advection–diffusion equation, but these often lack the ability to account for stochastic fluctuations and complex boundary conditions (Márquez-Romance et al., 2022; Kadanoff and Swift, 1968).

Stochastic models account for the inherent randomness in pollutant dispersion processes. These models, often formulated as stochastic differential equations, incorporate random perturbations to the advection-diffusion framework (Kerachian and Karamouz, 2007). They provide a probabilistic description of pollutant concentration, offering insights into the variability of diffusion processes (Pelosi et al., 2016, 2014).

However, the most suitable strategy consists of combining different modeling techniques and data in order to mitigate uncertainties and improve the accuracy of pollution assessments (Wing et al., 2020; Kerachian and Karamouz, 2007).

Recently, we developed a mathematical model of pollutant dispersion in river networks (Rizzello et al., 2023a,b) on two observation scales, a channel scale and a basin scale, through a complex system Master Equations, which allow for pollutant transport studies.

In this paper we describe the algorithm that we used in order to provide an iterative solution of the Master Equation. The algorithm has been implemented in two different software packages written, respectively, in the computer algebra systems Reduce and Mathematica.

Initially, we implemented the algorithm in Reduce. Indeed, Reduce's language is a derivative of Lisp, and it is very suitable for dealing with trees as the main data structure; indeed, trees are used to model river networks. Reduce is now free software, and it can be downloaded at Reduce project page (2023), thus making it a quick solution if there is an interest in doing a simulation. To this aim, we released our code in `GitHub` (Rizzello et al., 2024), so that our results are reproducible by any interested scientist. That could also be a starting point for future developments.

Subsequently, the software package was translated into Mathematica, in order to access the richer built-in graphical functions offered by this environment. Of course, the drawback is that the above computer algebra system is proprietary, and rather expensive (although quite common in the Engineering community).

Despite the syntactical and structural differences between these two environments, the two implementations behave approximately in the same way. We provide as an example a simulation on a complex river network, that was discussed in Rizzello et al. (2023a,b), in both implementations; of course, in the end we get the same results.

The algorithm's approach involves the representation of river networks as a tree, or a hierarchical set of nodes and links. This is implemented as nested lists. Each level of nesting corresponds to specific components, such as river streams, junctions, and pollutant sources, building up an abstract representation of our intuitive picture or of a real situation.

The Master Equation is peculiar of our model of the physics of pollutant advection. It expresses in a simplified way the physical principles of mass and momentum conservation. The Master Equation is calculated by means of recursive symbolic procedures which can descend nested lists in a natural way.

A key advantage of this algorithm and its implementation is its scalability, a valuable property in applications where networks are subject to continuous changes or expansions.

As the network grows or evolves over time, the symbolic procedures adapt seamlessly, ensuring efficient and accurate real-time generation of the equations for all nodes (Rizzello et al., 2023a,b). This adaptive approach allows for efficient manipulation of the nested lists, enabling easy updates and modifications of the domain properties in the software.

The Master Equation that holds at each node of the river network is in continuous time:

$$\rho(i,t+\Delta t)=\rho(i,t)+\sum_{j=1}^{N}P_{ji}\rho(j,t)-\sum_{j=1}^{N}P_{ij}\rho(i,t). \tag{1}$$

which can be rewritten on a discrete time grid: if n is an integer, Eq. (1) become:

$$\rho(i,t_0+(n+1)t)=\rho(i,nt)+\sum_{j=1}^{N}P_{ji}\rho(j,nt)-\sum_{j=1}^{N}P_{ij}\rho(i,nt). \tag{2}$$

where ρ is the physical quantity that we want to determine (a pollutant or sediment concentration), t_0 is the initial time, Δt is the size of the total time step, t is the discrete time, i and j the ith and jth node of the discretized network respectively, n the nth time step, N the final time step and P_{ij} the transition probability between nodes i and j.

The parameter P_{ij} can be a constant or a function of time. It can depend on the length of the stream, its geometry, diffusion coefficients, flow velocity and other quantities; it summarizes the physics of the advection process. It describes the momentum conservation while the Master Equation expresses the mass conservation (Rizzello et al., 2023a,b) between nodes of the river network.

At the basin scale, the river network is combined by several minimal networks, thus the phenomenon can be described by several local interactions. Referring to a local interaction, Eq. (1) (or on a discrete time grid (2)) is applied for each channel considering the minimal structure in which three or more channels are connected.

For example, considering the scheme in Fig. 1, Eq. (1) turns into his complete local form:

$$\rho_j(t_k)=\rho_j(t_{k-1})+P_{mj}\rho_m(t_{k-1})+P_{nj}\rho_n(t_{k-1})-P_{ji}\rho_j(t_{k-1})\ , \tag{3}$$

or equivalently,

$$\rho_j(t_k)=\rho_j(t_{k-1})-P_{ji}\rho_j(t_{k-1})+\sum_{l=n}^{m}P_{lj}\rho_l(t_{k-1})\ . \tag{4}$$

By an iterative application of the Master Equation at various nested levels, the model accommodates the interdependence in time of water flow, pollutant transport, and other hydraulics and hydrological variables. Solving the equations yields the temporal evolution of variables at different time intervals t on all individual nodes.

Practical applications of the proposed method include simulations of pollutant advection/dispersion in river networks, dynamics of biological systems, and modeling complex physical phenomena (Rizzello et al., 2023a,b).

The algorithm for the computation of the Master Equations is described in the next section, and its Reduce and Mathematica implementation can be found in the two subsequent sections. A meaningful example is programmed in both computer algebra system and is discussed in details.

3. The algorithm

It is well-known in Computer Science that any algorithm is expressed in a simpler way if a suitable data structure is chosen. The methodology is applied below on a theoretical case to explain how the algorithm works, but it has been applied to real cases at channel and basin scales (Rizzello et al., 2023a,b).

A flow chart of the algorithm is provided in Fig. 2 for a better understanding of the symbolic software packages that we developed so far (Rizzello et al., 2024).

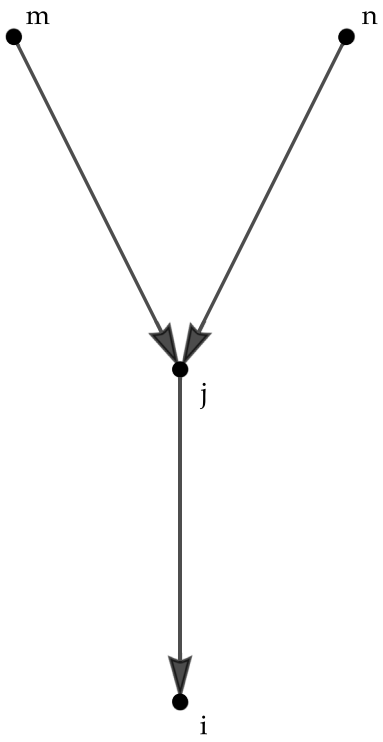


Fig. 1. Hortonian minimal structure for local reaction, *i, j, m, n,* are the nodes of the network.

The computational method begins with the symbolic representation of a given river network as a nested list, capturing its spatial and functional structure.

The river network can be mathematically represented as a tree graph, which is a set of connected streams. Its basic elements are nodes, streams, and channels. Nodes represent the junction points of two (or more) tributaries or the locations of sources. In general, the number of tributaries connected at a node is 2. In special cases, such as the presence of fractures or joint faults, this number can be greater. Streams (or edges) are the segments between two consecutive nodes, while channels are sequences of streams with the same order. The terminal node is referred to as the root, or outlet.

In general, the root corresponds to a closing section of hydrological–hydraulic interest. Thus, from a topological viewpoint, the number of nodes, n_n, is given by the sum of sources, n_s, junctions, n_g, and the root:

$$n_n=n_s+n_g+1\ . \tag{5}$$

The numbers of junctions, n_g, nodes, n_n, and streams, n_r, can be easily derived from the number of sources, n_s, as follows:

$$n_g=n_s-1\ , \tag{6}$$

$$n_n=2n_s\ , \tag{7}$$

$$n_r=n_n-1=2n_s-1\ . \tag{8}$$

These equations are pivotal in characterizing the domain discretization.

Assuming the Digital Elevation Model (DEM) is known, it is necessary to extract the river network that can subsequently be discretized into nodes and (connecting) arcs. For example, a DEM of a basin near the city of Mesa (California, USA) is shown in Fig. 3 with a resolution of 1 m × 1 m, obtained using the LiDAR optical sensor technology.

Specifically, with the D8 algorithm, drainage networks are produced by applying a threshold value to the flow accumulation data. Cells with a contributing area greater than a defined threshold are classified as part of the drainage network. The density of the drainage network increases with the threshold value of the number of cells n_c.

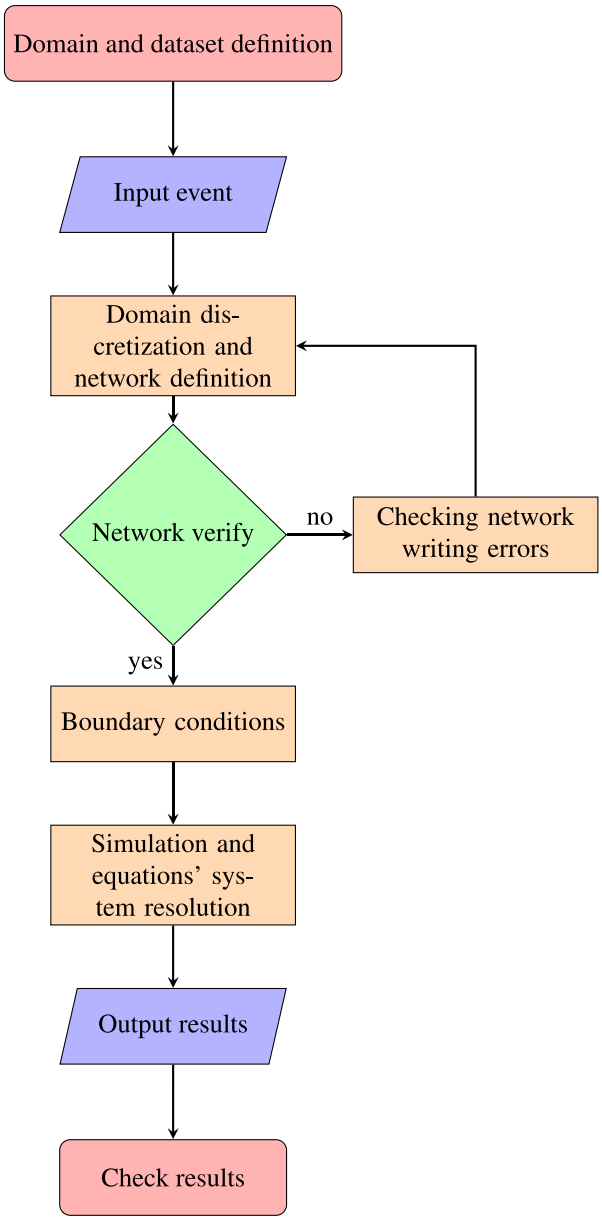


Fig. 2. Flow chart with a listing of the main parts of the Algorithm that computes the Master Equation and its solution.

The determination of an appropriate contributing area threshold is difficult, and needs to take into account the DEM resolution and terrain characteristics. Thus, the river network in our example is extracted using D8 algorithm with a threshold $n_c = 25000$ cells as depicted in Fig. 4.

It is very natural to describe the river network into nodes and arcs. Each level of nesting corresponds to a distinct spatial or functional element, providing a clear and intuitive representation.

More particularly, the river network can be modeled as a single list $\mathcal{L}$. The first element of the list is a node, denoted by n_1, and it represents the outlet node of the river network. The other elements of the list are lists. Each of these lists is a sub-network of the main network.

The structure of the algorithm is as follows.

- The first step consists into carrying out consistency checks on the list $\mathcal{L}$: the list must begin with a node n_1, and it should not contain repeated nodes or empty lists.

Fig. 3. Digital Elevation Model (DEM) of the basin near the city of Mesa in the State of California (USA). This DEM is obtained using the LiDAR optical sensor technology, with a resolution of 1 m × 1 m.

- After that, the initial conditions from which the computation will begin must be assigned. They can be defined at the initial time $t_0 = 0$. We shall define the values of transition probabilities, P_{ij}: they characterize the arcs joining pairs of nodes. Moreover, we shall define the density of the pollutant at specific source nodes; typically, at the boundary of the river network.
- Finally, the computation of the Master Equations from an initial time $t_0 = 0$ to a final time $t_n = n$ takes place. The tree structure makes it natural to program a recursive computation where Master Equations are iteratively applied at subsequent levels of nesting.
- The concentration results are stored in memory, so that it is possible to graph the trend of the physical quantities ρ as a function of instants t.

4. Reduce implementation

We developed a Reduce library containing the functions necessary to perform simulations. The library is available in our `GitHub` repository (Rizzello et al., 2024).

After some initialization commands in Reduce, the first fundamental routines of the library are:

```
validate_net1(net)
validate_net2(net)
```

Their purpose is to make sure that the input river network `net`, loaded as a list, has the proper syntax.

In particular, the first procedure checks if the main list `net` and its sub-lists all begin with nodes. The successor of the last node (in the sense of the lowermost node of the river basin) is denoted by `n0`; that is the first element of the data structure `net`. Next elements are the branches of `net`, which are lists, each beginning by a node. This recursively defines the data structure, and any input river network should be checked for consistency with this definition.

The second procedure checks if there are repeated nodes by calculating the list at zero `depth` and finding repeated elements.

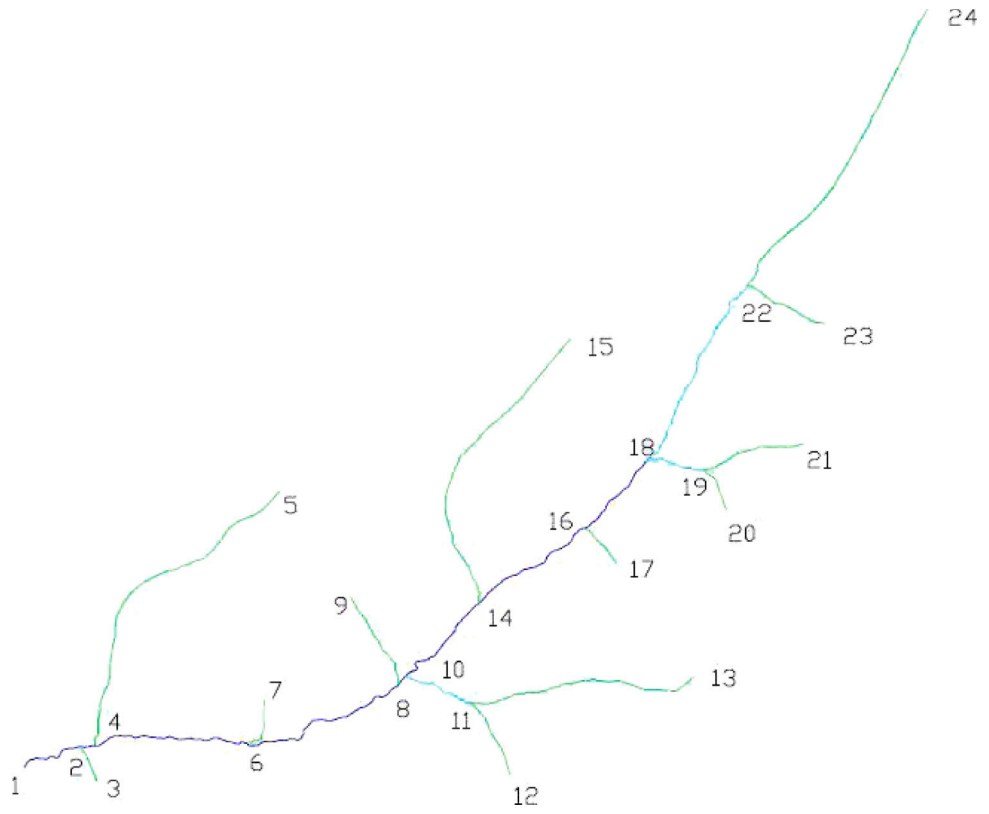


Fig. 4. Complex river network extracted using D8 algorithm with a threshold $n_c = 25000$ cells, discretized in nodes and arcs. The different colors defines a different Horton order of the streams.

The commands issue an error message if the input list does not comply with the criteria, and it is silent (no output) otherwise. These symbolic procedures are of fundamental importance because they check the presence of errors in the input variables before computing the Master Equations

The main routine of the software is

```
all_master_equation(net,rho_op,t0,n,tprob)
```

which takes as arguments the river network `net`, the operator `rho` defining the density of the physical quantity that is diffusing in the `net` at each time step, the initial time `t0`, the final time, `n`, and the transition probabilities between consecutive nodes `tprob`.

The main routine uses some subroutines to calculate the Master Equations through a cycle for each node of the river network; for clarity, it is written below:

```
algebraic procedure all_master_equation(net,rho,t0,n,tprob);
  begin
    scalar unfolded_net,l_branches,downknot,iter_branches,
      temp_bran,temp_iter_bran,cnt;
    cnt:=0;
    unfolded_net:=process_net(net);
    l_branches:=process_unfolded_net(unfolded_net,length(unfolded_net));
    iter_branches:=l_branches;
    while not(iter_branches={}) do
      <<
        temp_bran:=first(iter_branches);
        temp_iter_bran:=rest(iter_branches);
downknot:=find_downknot(temp_bran,temp_iter_bran);
local_master_equation(downknot,temp_bran,rho,t0,n,tprob);
iter_branches:=temp_iter_bran
      >>
  end;
```

The procedure uses two subroutines to prepare the main list for the computation of Master Equations:

```
process_net(net)
process_unfolded_net(unfolded_net,length(unfolded_net))
```

These procedures rewrite the main net. The first one opens the list as tree, into the list of its branches or sub-lists in a recursive way, yielding the list `unfolded_net`. The second one processes the `unfolded_net` to return the list of branches reversed. This last step is necessary to find the subnode or 'downknot' of each node of the network with the subroutine

```
find_downknot(temp_bran,temp_iter_bran)
```

More precisely, the above procedure finds for each branch or sub-list the hierarchically superior element, or equivalently, the immediately lower node in the river network.

Finally the Master Equation is written by means of the following procedure:

```
local_master_equation(downknot,temp_bran,rho,t0,n,tprob)
```

for each branch structure considering the sub-node (`downknot`) and their upper nodes (`temp_bran`) connected to it. In the procedure, `tprob` is the operator that yields the probability P_{ij} of transition between each couple of nodes. Such an operator should be defined in the initialization of the program, and it models the physical characteristics (with respect to diffusivity) of the basin at all nodes.

The Master Eq. (2) is computed by the above procedure by means of consecutive iterations, starting from the values at the initial time `t0`.

The Reduce substitution mechanism will automatically replace the values of the operators `rho` and `tprob` when necessary, at each time step and given nodes.

4.1. Reduce example

In this Section we describe as an example a diffusion process in the complex river network that we introduced in Section 3.

We start by initializing the density operator ρ:

```
operator rho;
for all t,s let rho(0,t,s)=0;
```

The second line is needed in order to set up properly the Master Equation for the root node of the river network. Indeed, 0 stands for a virtual root node where all values of the operator are null; the flow stops at the root node.

Then, it is necessary to initialize the set of probabilities of transition. Again, the virtual root node has 0 probability of transition:

```
operator tprob;
for all m let tprob(m,0)=0;
```

The next step is to load the library file:

```
in "MasterEqRiverNetwork.red";
```

The library contains all the routines to check the tree that models the network and to compute the Master Equations. Starting from the river network in Fig. 4, where a hierarchy between branches has been deduced by means of the geographic data, it is necessary to encode the network domain into the computer algebra system. The command defining the example river network as a list is:

```
net:={n1,{n2,{n3},{n4,{n5},{n6,{n7},{n8,{n9},{n10,{n11,{n12},{n13}},{n14,
{n15},{n16,{n17},{n18,{n19,{n20},{n21}},{n22,{n23},{n24}}}}}}}}}};
```

The list needs to be checked and validate using two routines:

```
validate_net1(net);
validate_net2(net);
```

The quantity ρ is 0 at every node except nodes 5, 15 and 21. This is set by computing the list of all nodes of depth one and then assigning values of `rho` on every node at the initial time `t0`:

```
flatnet=mkdepth_one(net);
for s:=1:length(flatnet) do rho(part(net,s),t0,0):=0;
rho(n5,t0,0):=5;
rho(n15,t0,0):=4;
rho(n21,t0,0):=6;
```

The transition probability is further initial data. In this example they are defined after being generated in a random way.

```
tprob(n2,n1):=0.69;
tprob(n3,n2):=0.82;
tprob(n4,n2):=0.94;
tprob(n5,n4):=0.30;
tprob(n6,n4):=0.67;
tprob(n7,n6):=0.84;
tprob(n8,n6):=0.47;
tprob(n9,n8):=0.68;
tprob(n10,n8):=0.97;
tprob(n11,n10):=0.71;
tprob(n12,n11):=0.79;
tprob(n13,n11):=0.39;
tprob(n14,n10):=0.56;
tprob(n15,n14):=0.32;
tprob(n16,n14):=0.71;
tprob(n17,n16):=0.83;
tprob(n18,n16):=0.67;
tprob(n19,n18):=0.83;
tprob(n20,n19):=0.81;
tprob(n21,n19):=0.59;
tprob(n22,n18):=0.41;
tprob(n23,n22):=0.81;
tprob(n24,n22):=0.31;
```

The Master Equations will be computed at each node of all valid sub-nets and at all times. Initially, at a given time t_k, for each node n_i the software calculates the only adjacent downstream node that follows n_i (from upstream to downstream). The process is iterated for every node, and eventually all the Master Equations at all nodes of every sub-net are then generated. This yields the value of the density `rho` at every node and at the next time step t_{k+1}.

The input command line is:

```
tn:=15;
for i:=1:tn do all_master_equation(net,rho,t0,i,tprob);
```

The density values are stored in the operator of the form:

```
rho_op(ni,t0,k);
```

where `ni` is the ith node, `t0` is the initial time and `k` is the index of the discrete time step.

In order to compute the Master Equations at a discrete time t_k in the grid, all previous Master Equations at all times 1, ...,`k-1` have to be already computed. The Reduce substitution mechanism will automatically replace the values of the operator `rho` at previous times when necessary. Finally, the results of ρ for each node can be collected and plotted using the `gnuplot` library.

5. Mathematica implementation

Here we introduce the Mathemathica implementation of the dispersion model on complex river networks. It consists of a translation of what is written in Lisp code; the main aim is to make all the capabilities by the Mathematica environment available for post-processing of the output of the Master Equation. In this way, we have been able to introduce few improvements with respect to the Reduce implementation; we discuss them below.

The library is written as a series of procedures divided in three parts according to their purpose. Generally, the variables of iterations are i and j, while the variables s and t are used for storing data. The first part contains initializing procedures. The second part contains the procedures that verify the river network, while the last part concerns the computing procedures of the Master Equations.

After a quite straightforward initialization, the river network verification routines are implemented in the procedure `VerifyNetwork[z]`:

```
VerifyNetwork[z_]:=(If[CheckList[z],
(Print[Style[``The network is a list.'', RGBColor[0,1,0]]];
CheckNodes[z];
CheckRepeatedNodes[z];
CheckUnpairedSourceNodes[z];),
Message[VerifyNetwork::Error1]];
Print[
"Checks concluded! Verification of the river network is complete!"])
VerifyNetwork::Error1="The network must be a list.";
```

In this routine the river network is defined as `z` and the subroutines follow the same pattern as in the Reduce implementation. Namely, they check, respectively, if the network presents some empty sub-lists or repeated nodes and additionally a check on the source nodes is performed, counting the brackets of the network and checking if they are closed correctly. In particular `CheckUnpairedSourceNodes[z]` loops over the entire list by storing in memory a list which contains

all the correctly written source nodes, checking also that there are no unpaired nodes.

An additional feature has been introduced in the Mathematica implementation. More precisely, the computational domain represented by the river network as a list can be further discretized on the basis of real lengths (to be defined in the environment) by adding intermediate nodes and arcs in order to evaluate the physical quantity ρ at internal points which are not defined in the initial domain. The routine is:

```
CreateNestedNetwork[net,Lb,P]:=(Lb=Lb;
ComputeP[net,RatioL];
ComputeP[net,AdditionalSubLengths];
ComputeP[net,AdditionalNodes];
Couples = ComputeP[net,couple];
Print["The base length is: ",Lb];
Print["The number of additional nodes to ",
Length[Flatten[net]],"existing nodes is:",
(1 + Total[ComputeP[net,AdditionalNodes]]),". The total number of nodes is ",
Length[Flatten[net]]+1+
Total[ComputeP[net,AdditionalNodes]],"."];
Print[
"Furthermore, P of each branch is the same of the initial branch."];
Do[CreatAdditionalLists[First[couples[[i]]],Last[couples[[i]]],P],
{i,1,Length[couples]}];
Clear[subnode];
NestingNetwork[net];
Print["The depth of the nesting nested network is: ",Depth[nestdnet],"."];
nestdnet)
```

The above routine is made by different procedures. In particular, after defining the lengths between each couple of nodes, one procedure computes additional sublists (`AdditionalSubLengths`) and additional nodes (`AdditionalNodes`) that can be added to the main river network by increasing the domain discretization step. For each branch of the network, the newly created sublists inherit the same transition probability of the branch.

Again, the most important part of the software lies in the computational routines below. They have been written in a different way with respect to the Reduce implementation in order to speed up the computation. Different pointers are necessary to move inside the network and store the position of nodes at which the computation is made. This is performed by the following routine:

```
LocalNodes[net,j]:=(
MEDownPointer[net,j];
MELUpPointer[net,j];
MEListDownPointer[net,j];MESubnodeFinder[net,j];
MEListUpPointer[net,j];
MeSupernodesFinder[net,j];)
Truncation[x]:=If[x<truncation,0,x];
```

Two pointers are used in order to move to higher or lower depths into the network list and results are stored into two lists. This feature is used in the computing routine of Master Equations listed below.

```
ComputeMasterEquations[net,rho,m,n,P]:=(truncation=10^-5;
progression=PrintTemporary["Computing:",Dynamic[status]];
Print["Thenecessarytimeis",First[AbsoluteTiming[
(Clear[flatnet];flatnet=Rest[Flatten[net]];Lcycle=Length[flatnet];
Clear[MEsubnode];Clear[MEsupernodes];
MEsubnode={};
MEsupernodes={};
Do[(LocalNodes[net,j]),{j,1,Lcycle}];
firstsupernodes=Table[First[Rest[net][[k]]],{k,1,Length[Rest[net]]}];
matrixPsubnodes=Table[P[flatnet[[w]],MEsubnode[[w]]],{w,1,Lcycle}];
matrixPsupernodes=Table[(If[MEsupernodes[[w]]=={},0,
Table[P[MEsupernodes[[w]][[q]],flatnet[[w]]],
{q,1,Length[MEsupernodes[[w]]]}]),{w,1,Lcycle}];
matrixrhosupernodes=
Table[(If[MEsupernodes[[w]]=={},0,Table[rho[MEsupernodes[[w]][[q]],m-1],
{q,1,Length[MEsupernodes[[w]]]}]),{w,1,Lcycle}];
matrix=Table[rho[flatnet[[w]],m-1],{w,1,Lcycle}];
Do[
(
(rho[First[net],i]=rho[First[net],i-1]+Sum[rho[firstsupernodes[[c]],i-1]*
P[firstsupernodes[[c]],First[net]],{c,1,Length[firstsupernodes]}];);
matrixproduct=matrixrhosoupernodes*matrixPsupernodes;
matrixproduct=Map[Total,matrixproduct];
matrix=matrix-matrix*matrixPsubnode+matrixproduct;
matrix=Map[Truncation,matrix];
(Do[(rho[flatnet[[j]],i]=matrix[[j]];),{j,1,Lcycle}];
matrixrhosupernodes=Table[(If[MEsupernodes[[w]]=={},0,
Table[rho[MEsupernodes[[w]][[q]],i],Table[rho[MEsupernodes[[w]][[q]],i],
Table[rho[MEsupernodes[[w]][[q]],i],
{q,1,Length[MEsupernodes[[w]]]}]),{w,1,Lcycle}];
status=PercentForm[N[Round[((i-m)/(n-m)),0.0001],4]];)
),{i,m,n}];
(NotebookDelete[progression];
Clear[status];
Clear[progression];
Clear[truncation];
Clear[Lcycle];
Clear[matrix];
Clear[matrixrhosupernodes];
Clear[matrixPsubnode];
Clear[matrixPsupernodes];
Clear[matrixproduct];
);];]]);
,"seconds."]);
```

Storing the pointers allows us to speed up the above procedure through matrix operations. Indeed, considering Eq. (2) written for the ith node, each term is stored forming a matrix if the whole network is considered. Then Eq. (2) transforms in a matrix equation at the fixed time step. The computation is performed for all the time steps considered. From a computing point of view, the main difference between Reduce and Mathematica is that the computation of the Master Equations in Mathemathica is performed using matrix operations resulting in a reduced computational time.

Lastly, some graphical routines are introduced to represent the network as a hierarchical tree. A schematization of the network of Fig. 4 is shown below (see Fig. 5).

5.1. Mathematica example

In this section we recompute the same example that we computed above using the Reduce library.

The first step is to load the library through the following command lines.

```
NotebookEvaluate[StringJoin[ToString[NotebookDirectory[]], "library.nb"]]
```

The second step is to define the river network as a list.

```
net={n1,{n2,{n3},{n4,n5},{n6,{n7},{n8,{n9},{n10,{n11,{n12},{n13}},{n14,
{n15},{n16,{n17},{n18,{n19,{n20},{n21}},{n22,{n23},{n24}}}}}}}}}};
```

The network list needs to be checked also in Mathematica:

```
VerifyNetwork[net];
```

It checks multiple conditions; in particular it verifies that the input river network is not an empty list, that it does not contain repeated nodes and it checks if there are unpaired nodes by counting the number of brackets. If the test is passed the output is silent, errors will be showed otherwise.

After passing the previous checks, the computational domain can be further discretized on the basis of real lengths (to be defined in the Mathematica environment) by adding intermediate nodes and arcs in order to evaluate the physical quantity ρ at internal points not defined in the initial domain.

As in Reduce example the initial conditions need to be defined. The quantity ρ is 0 at every node except nodes 5, 15 and 21.

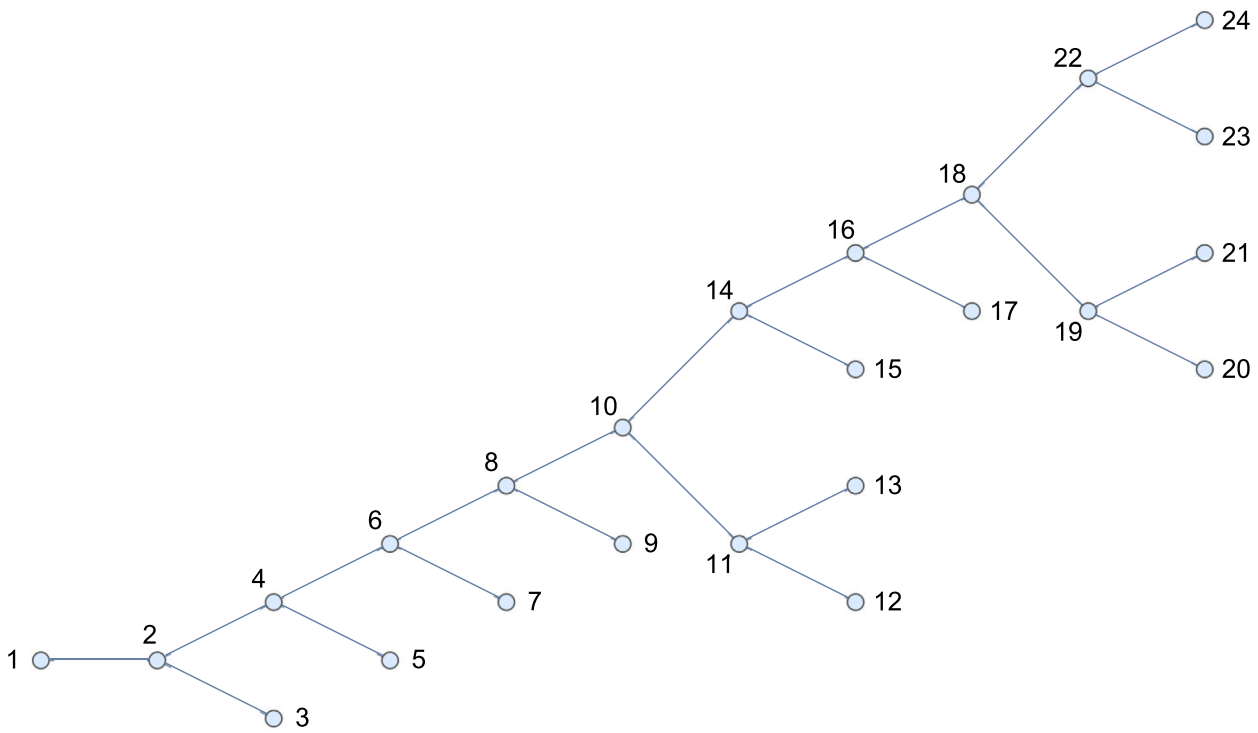


Fig. 5. Hierarchical tree representation of the Mesa river network. This scheme consists of 24 nodes and 23 streams.

```
flatnet = Flatten[net];
Do[(rho[flatnet[[i]],0]=0;), {i,1,Length[flatnet]}];
rho[n5,0]=5;
rho[n15,0]=4;
rho[n21,0]=6;
```

Note that the dependence on t_0 has been removed in the definition of ρ (or in Mathematica `rho[ni,i]`, where `ni` is the ith node and i is the ith time step). Then, the transition probabilities P_{ij} are defined.

```
P[n2,n1]=0.69;
P[n3,n2]=0.82;
P[n4,n2]=0.94;
P[n5,n4]=0.30;
P[n6,n4]=0.67;
P[n7,n6]=0.84;
P[n8,n6]=0.47;
P[n9,n8]=0.68;
P[n10,n8]=0.97;
P[n11,n10]=0.71;
P[n12,n11]=0.79;
P[n13,n11]=0.39;
P[n14,n10]=0.56;
P[n15,n14]=0.32;
P[n16,n14]=0.71;
P[n17,n16]=0.83;
P[n18,n16]=0.67;
P[n19,n18]=0.83;
P[n20,n19]=0.81;
P[n21,n19]=0.59;
P[n22,n18]=0.41;
P[n23,n22]=0.81;
P[n24,n22]=0.31;
```

The Master Equations can be computed:

```
t1=1;
tn=15;
ComputeMasterEquations[net,rho,t1,tn,P];
```

Through this command the recursive symbolic procedures are thus applied to dynamically generate Master Equations for each node. Note that the procedure can be stopped at a specified time t_k, and then it can be resumed. Results are stored as `rho[ni,ti]`. Below there are the outputs of ρ for some nodes of the network in a graphical representation.

Fig. 6 shows the output. In particular, blue shows the n1 closure node representing the cumulative ρ curve, while the other curves represent the pollutant wave propagating from upstream to downstream in the considered streams. As this wave propagates downstream from the source nodes to the closure section, a temporal shift of the concentration curves at the intermediate nodes can be observed.

6. Geographical area and study case

The geographic area under study is a region of significant environmental importance, where maintaining the fragile balance between human activities and ecosystem health is a pressing issue. The focus of the study is the Maumee River Watershed, located in the states of Ohio and Indiana (USA), known for its distinctive geographical and hydrological features. This watershed is one of the largest in the Great Lakes region. It flows through the U.S. Midwest, originating in northeastern Indiana and continuing into northwestern Ohio before emptying into Lake Erie.

The Maumee River Basin holds historical significance as a key transportation route for Native American tribes and European settlers. However, the river's pollution history shifted in the 19th and early 20th centuries due to rapid industrialization and urbanization in the region (Siman and Niewiarowski, 2023; Kaatz, 1955). As industries like manufacturing, agriculture, and shipping grew, the discharge of pollutants into the river increased accordingly.

As in previous works (Rizzello et al., 2023a,b) the domain is the catchment area of the Maumee River's Basin in the Ohio State (USA),

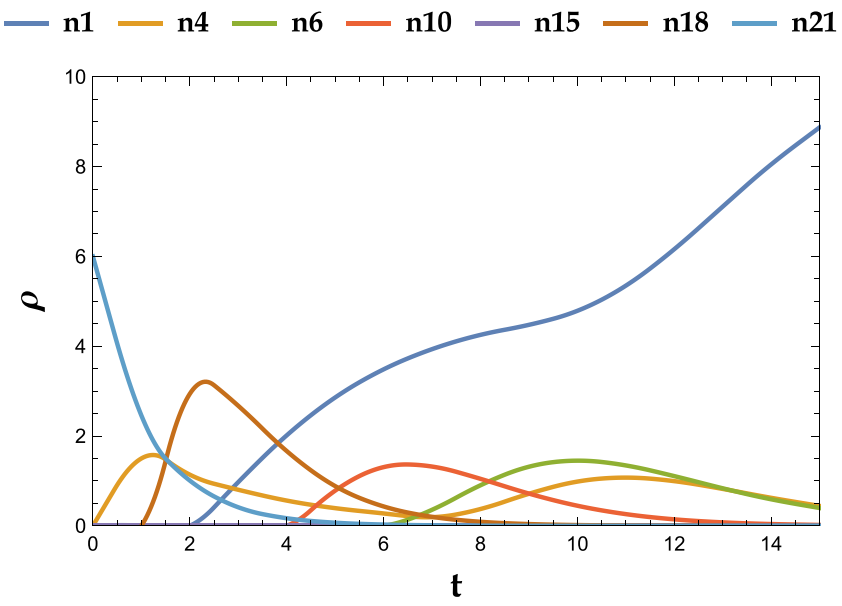


Fig. 6. An example of concentration curves obtained from a synthetic simulation carried out on the Mesa river network. The simulation concerns fictitious cases of ρ for nodes 1, 4, 6, 10, 15, 18, 21.

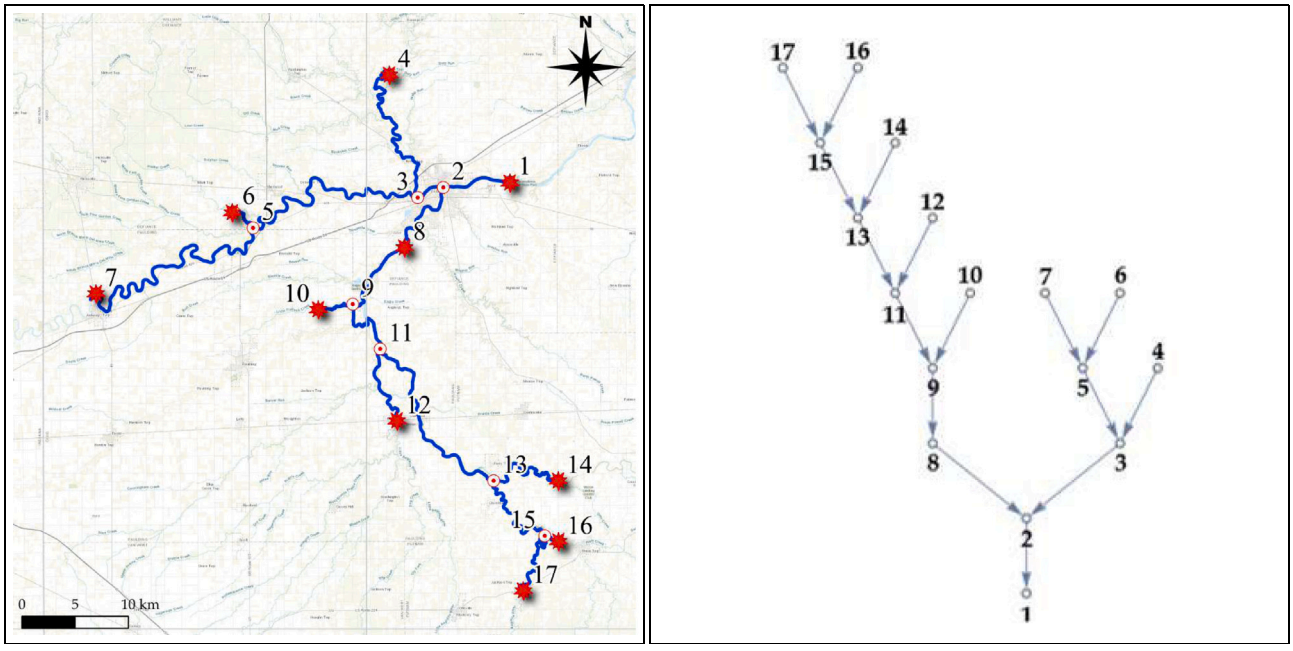


Fig. 7. On the left, the geographical network area with the water courses; the gauging stations are represented as red dots and the junction points between channels are represented by white dots. On the right, the discretized network domain.

at basin scale, represented in Fig. 7. The data, provided by the USGS, range from mid-April to mid-May 2017.

This study area features a diverse landscape shaped by a complex interaction of natural elements and human activities, making it an ideal setting for examining the dynamics of pollutant advection.

As a consequence of its environmental significance, the region has been equipped with monitoring stations by the USGS (United States Geological Survey) to track water quality parameters (Shoda et al., 2015). These stations are strategically located to cover a range of water bodies, from major rivers to smaller streams and reservoirs.

7. Results and discussion

The article outlines the step-by-step implementation of the recursive Master Equation generation in two different computer algebra environments. Simulating pollutants within a river network involves simulating a combination of hydrodynamical, chemical, and biological processes. The symbolic computation approach enables to address a wide range of scenarios by adapting the symbolic procedures to specific node properties and network structures. In this work we considered a theoretical example as well as a real case. The real case was considered

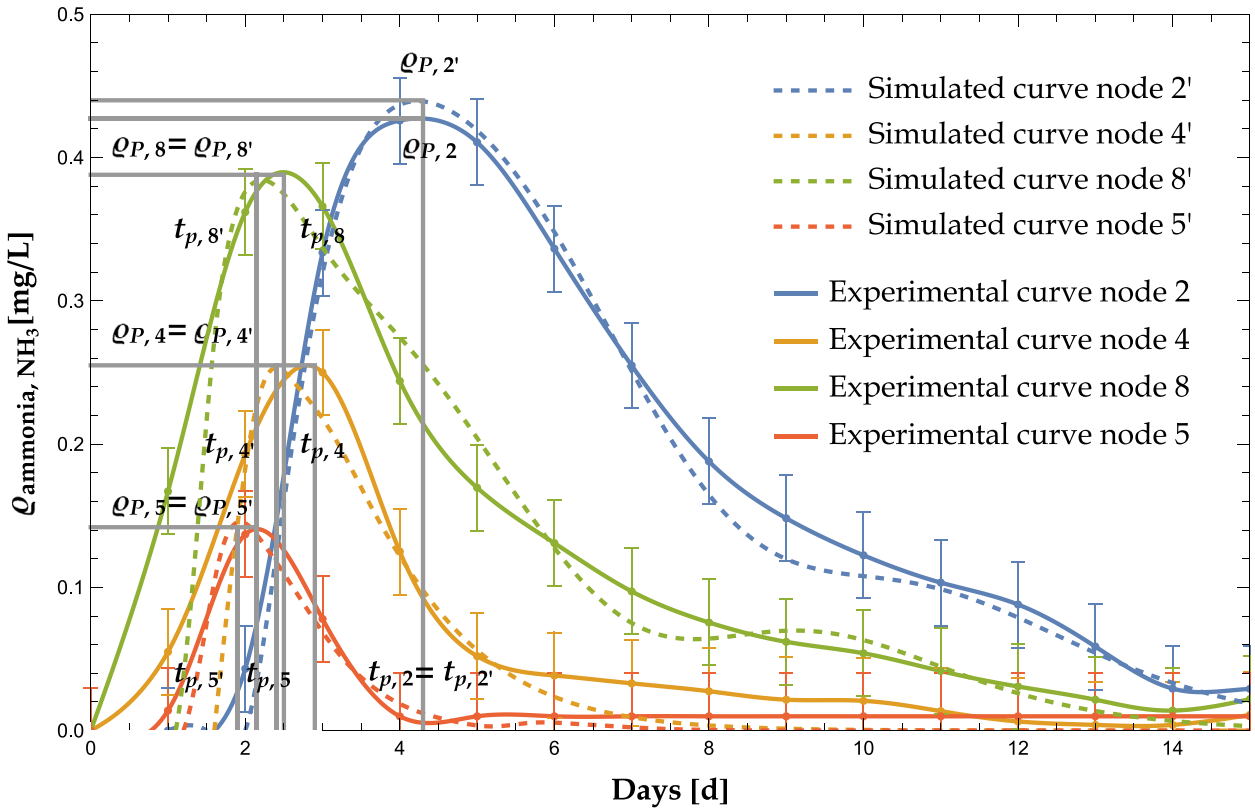


Fig. 8. Development of ammonia concentrations (ρ_{NH_3}) versus time. The peaks and their respective times are highlighted.

in Rizzello et al. (2023a,b) to address the dispersion analysis of some contaminants within the catchment area of the Maumee river basin, at channel and basin scale. The model has provided interesting results in terms of peak time and maximum solute concentration.

As an example, the results of the real case are reported below. For the ammonia pollutant NH_3, we analyzed the concentration peaks of source nodes (in this case, source nodes are 4, 5,8) and closure node is 2. Results are shown in Table 1, where we show the transition probabilities $P*$ and P_{ij}, the time peak t_p, the concentration of ammonia ρ, the time-lag, the error between observed and simulated peaks ($\prime$), the determination coefficient R^2 of the simulated data ($\prime$) on the nodes, 2, 4, 5, 8, respectively. While the development of ammonia concentrations, ρ_{NH_3}, versus time, t, is shown in Fig. 8. In the observed curves, the error bars are obtained directly from the data provided by the USGS.

In the first instance, for each couple of arcs (streams) it is assigned the same transition probability, $\boldsymbol{P}^*$, which is the transition probability of first attempt, and is assigned to 0.65. The optimal $\boldsymbol{P}^*$ value is obtained in an iterative manner by finding a simulated concentration curve that comes closest to the experimental one and thus minimizing the errors between the concentration peaks and the respective time lags (Rizzello et al., 2023a,b).

Then, the second step is to adjust the local transition probabilities, specifically $P_{8,2}$ was adjusted to 0.40, instead $P_{5,3}$ was adjusted to 0.43 and $P_{4,3}$ was adjusted to 0.30. The determination coefficient, R^2, for the simulated curves was determined through the following relation:

$$R^2 = 1 - \frac{SS_{\text{res}}}{SS_{\text{tot}}}, \tag{9}$$

where, SS_{res} is the residual sum of squares between the observed and simulated data, and SS_{tot} is the total sum of squares between the observed and simulated data. The results showed in most cases values of R^2 very close to unity, with the only exception of the simulated curve $8'$ with a value 0.765. All simulated curves were obtained with third-order interpolating functions.

The relative error (%) between the observated and simulated peak curves defined as $\frac{|\rho_N - \rho_{N'}|}{\rho_N}$ (where N is the generic node) is evaluated for the nodes under consideration. Thus, for the outlet node 2 it is 3.3%. For the remaining nodes it was between 0.2% and 1.8%. In addition, there is a time-lag of 0.50 days, 0.60 days and 0.05 days, (about 12 h, 14.4 h and 1 h) for source node 4, 5, 8 respectively.

In particular, the results for the analyzed event are shown for all the nodes in Fig. 8, with the closure node 2 in blue, and the source nodes 4, 8 and 5 in orange, green and red respectively. However, it is important to note that, overall the time lag is less than 1 day and the peak concentrations have low relative peak errors. The model, when used appropriately (and also in conjunction with other modeling approaches and field data), can offer valuable insights into pollutant behavior and aids in developing effective pollution management strategies for reducing the impact on the environment.

8. Conclusion and outlook

In this work a computational method was defined by means of computer algebra procedures to simulate pollutant diffusion in river networks. The method is based on the use of Master Equations. Representing river networks as trees simplifies the derivation of governing equations for pollutant transport, including mass balance equations that account for sources, sinks, and transport within the network. Two software libraries were created for this purpose. The libraries were written using the Reduce and Mathematica computer algebra systems, respectively; moreover, two theoretical examples were provided.

We obtained results in good agreement with existing data (when available). We observed 4 experimental events provided by the USGS, ranging from mid-April to mid-May 2017, in several different measurement stations on the Maumee river basin in the Ohio state (USA). The precision of the numerical results was assessed through the determination coefficient R^2, whose value turned out to be close to 1 in most of the cases that we analyzed.

Table 1
Summary of the results for ammonia solute NH_3. In the table we show the transition probabilities $P*$ and P_{ij}, the time peak t_p, the concentration of ammonia ρ, the time-lag, the error between observed and simulated peaks, the determination coefficient R^2 of the simulated data on the nodes, 2, 4, 5, 8, respectively.

Transition probabilities	$P^* = 0.65,\ P_{8,2} = 0.40,\ P_{5,3} = 0.43,\ P_{4,3} = 0.30$				
Curve	t_p [d]	ρ_{NH_3} [mg/L]	Time-lag [d]	Error %	R^2
Experimental data node 2	4.30	0.42	–	–	–
Experimental data node 4	2.90	0.26	–	–	–
Experimental data node 5	2.50	0.14	–	–	–
Experimental data node 8	2.1	0.39	–	–	–
Simulated data node 2'	4.30	0.44	0.0	3.3	0.985
Simulated data node 4'	2.40	0.26	0.5	0.2	0.954
Simulated data node 5'	1.90	0.14	0.6	1.8	0.941
Simulated data node 8'	2.15	0.39	0.05	1.0	0.765

The above analysis show that our model (and our software), in spite of its simplicity, is able to reproduce observed pollution phenomena. However, in our opinion, our model, based on the Master equation, and our computer algebra methods could have a much wider range of applications in the Environmental and Hydraulic Engineering. Some of them are:

- in view of its simplicity, our model (and the software packages) could be integrated in most of the professional hydraulic modeling and GIS softwares;
- the model and the software analysis can be extended to much more complex river networks, with large Hortonian structures, enhancing the understanding of pollutant transport processes across various geomorphological settings. Indeed, technically speaking, our software scales very well to basins with a high number of nodes and streams;
- the model and the software packages also facilitate the exploration of potential differences and similarities between river networks, providing valuable insights for water quality management and environmental protection strategies on a broader scale (Rizzoli and Young, 1997; Harrison et al., 2023; Villa et al., 2009; Argent et al., 2009);
- the model and the software packages could be used to compare seasonal events across multiple years, aiming to identify dependencies in transport phenomena, such as seasonal average velocity or total water mass flowed;
- our software could be easily adapted to the analysis of the advection of pollutants over water distribution networks. The real-time analysis capability of our software would allow for almost instant answers about the possible diffusion of pollutants in networks of crucial social importance like acqueducts.

In conclusion, we think that our approach has interesting perspectives that we will explore in future research works.

CRediT authorship contribution statement

Samuele De Bartolo: Validation, Supervision, Methodology, Data curation, Conceptualization. **Gaetano Napoli:** Validation, Conceptualization. **Stefano Rizzello:** Validation, Supervision, Software, Methodology, Formal analysis, Data curation, Conceptualization. **Raffaele Vitolo:** Validation, Supervision, Software, Data curation, Conceptualization.

Software and data availability

Software name: MasterEqSoftware.
Developers: S. De Bartolo, G. Napoli, S. Rizzello, R. Vitolo.
First year available: 2024.
Hardware requirements: PC/Mac.
Software requirements: Reduce©, Mathematica©.
Program language: Reduce Lisp (RLisp), Mathematica© language.
Availability: https://github.com/stefano-rizzello/MasterEqSoftware.git
Size of archive: 0.3 Mb.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgments

The work of S.R. has been funded by the H@B Project F/340063/01-02/X59 - CUP: B89J24002570005 of the University of Salento, Italy. The work of G. N. and R.V. has been funded by the MIUR (Italian Ministry of Education, University and Research), Italy Project PRIN 2020, "Mathematics for Industry 4.0", Project No. 2020F3NCPX, and partially supported by GNFM of Italian INdAM, Italy. R.V. acknowledges the financial support of the INFN, Italy through the research project MMNLP and the financial support of Department of Mathematics and Physics 'E. De Giorgi' of the Università del Salento, Italy.

Data availability

Data/code are available at https://github.com/stefano-rizzello/MasterEqSoftware.

References

Alley, E.R., 2007. Water Quality Control Handbook. McGraw-Hill Education, New York.
Altenburger, R., Ait-Aissa, S., Antczak, P., Backhaus, T., Barceló, D., Seiler, T.B., Brion, F., Busch, W., Chipman, K., de Alda, M.L., et al., 2015. Future water quality monitoring. Adapting tools to deal with mixtures of pollutants in water resource management. Sci. Total Environ. 512, 540–551.
Ani, E.-C., Wallis, S., Kraslawski, A., Agachi, P.S., 2009. Development, calibration and evaluation of two mathematical models for pollutant transport in a small river. Environ. Model. Softw. 24 (10), 1139–1152.
Argent, R.M., Perraud, J.-M., Rahman, J.M., Grayson, R.B., Podger, G.M., 2009. A new approach to water quality modelling and environmental decision support systems. Environ. Model. Softw. 24 (7), 809–818.
Badrzadeh, N., Samani, J.M.V., Mazaheri, M., Kuriqi, A., 2022. Evaluation of management practices on agricultural nonpoint source pollution discharges into the rivers under climate change effects. Sci. Total Environ. 838, 156643.
Barati Moghaddam, M., Mazaheri, M., MohammadVali Samani, J., 2017. A comprehensive one-dimensional numerical model for solute transport in rivers. Hydrol. Earth Syst. Sci. 21 (1), 99–116.
Benedini, M., Tsakiris, G., 2013. Water Quality Modelling for Rivers and Streams. Springer Science & Business Media.
Botter, G., Bertuzzo, E., Rinaldo, A., 2010. Travel time distributions, soil moisture dynamics and the old water paradox. In: AGU Fall Meeting Abstracts. Vol. 2010, pp. H51B–0880.
Botter, G., Bertuzzo, E., Rinaldo, A., 2011. Catchment residence and travel time distributions: The master equation. Geophys. Res. Lett. 38 (11).
Cárdenas, S., Márquez, A., Guevara, E., 2023. Diffusion–advection process modeling of organochlorine pesticides in rivers. J. Appl. Water Eng. Res. 11 (1), 1–22.

Chatwin, P., Allen, C., 1985. Mathematical models of dispersion in rivers and estuaries. Annu. Rev. Fluid Mech. 17 (1), 119–149.
Costabile, P., Costanzo, C., De Bartolo, S., Gangi, F., Macchione, F., Tomasicchio, G.R., 2019. Hydraulic characterization of river networks based on flow patterns simulated by 2-D shallow water modeling: scaling properties, multifractal interpretation and perspectives for channel heads detection. Water Resour. Res. 55 (9), 7717–7752.
Costabile, P., Costanzo, C., Lombardo, M., Shavers, E., Stanislawski, L.V., 2024. Unravelling spatial heterogeneity of inundation pattern domains for 2d analysis of fluvial landscapes and drainage networks. J. Hydrol. 632, 130728.
De Bartolo, S., Dell'Accio, F., Frandina, G., Moretti, G., Orlandini, S., Veltri, M., 2016. Relation between grid, channel, and peano networks in high-resolution digital elevation models. Water Resour. Res. 52, 3527–3546.
De Bartolo, S., Dell'Accio, F., Veltri, M., 2009. Approximations on the peano river network: Application of the horton-strahler hierarchy to the case of low connections. Phys. Rev. E 79 (2), 026108.
Deng, Z.-Q., Jung, H.-S., 2009. Scaling dispersion model for pollutant transport in rivers. Environ. Model. Softw. 24 (5), 627–631.
Duarte, A.A., Boaventura, R.A.R., 2008. Dispersion modelling in rivers for water sources protection, based on tracer experiments: case studies.. Academia.
Harrison, P.A., Beauchamp, K., Cooper, J., Dickie, I., Fitch, A., Gooday, R., Holloway, M., Holman, I.P., Hunt, M., Jones, L., et al., 2023. An adaptable integrated modelling platform to support rapidly evolving agricultural and environmental policy. Environ. Model. Softw. 169, 105821.
Kaatz, M.R., 1955. The black swamp: a study in historical geography. Ann. Assoc. Am. Geogr. 45 (1), 1–35.
Kachiashvili, K., Gordeziani, D., Lazarov, R., Melikdzhanian, D., 2007. Modeling and simulation of pollutants transport in rivers. Appl. Math. Model. 31 (7), 1371–1396.
Kadanoff, L.P., Swift, J., 1968. Transport coefficients near the critical point: A master-equation approach. Phys. Rev. 165 (1), 310.
Kerachian, R., Karamouz, M., 2007. A stochastic conflict resolution model for water quality management in reservoir–river systems. Adv. Water Resour. 30 (4), 866–882.
Márquez-Romance, A., Cárdenas-Izaguirre, S., Guevara-Pérez, E., Pérez-Pacheco, S., 2022. Calibration of diffusion-advection process models for organochlorine pesticides in a tropical river. Environ. Qual. Manage. 31 (4), 403–424.
Mazaheri, M., Mohammad Vali Samani, J., Samani, H.M.V., 2015. Mathematical model for pollution source identification in rivers. Environ. Forensics 16 (4), 310–321.
Novotny, V., 1994. Water Quality: Prevention, Identification and Management of Diffuse Pollution. Van Nostrand-Reinhold Publishers..
Olowe, K.O., Kumarasamy, M., 2021. A mathematical model development for simulating nitrate pollutant transport along a river. Int. J. Eng. Res. Afr. 57, 149–168.
Pelosi, A., Parker, G., Schumer, R., Ma, H.-B., 2014. Exner-based master equation for transport and dispersion of river pebble tracers: Derivation, asymptotic forms, and quantification of nonlocal vertical dispersion. J. Geophys. Res.: Earth Surf. 119 (9), 1818–1832.
Pelosi, A., Schumer, R., Parker, G., Ferguson, R., 2016. The cause of advective slowdown of tracer pebbles in rivers: Implementation of exner-based master equation for coevolving streamwise and vertical dispersion. J. Geophys. Res.: Earth Surf. 121 (3), 623–637.
Reduce project page, 2023. The reduce project page at sourceforge. https://sourceforge.net/projects/reduce-algebra/.
Rinaldo, A., Gatto, M., Rodriguez-Iturbe, I., 2018. River networks as ecological corridors. A coherent ecohydrological perspective. Adv. Water Resour. 112, 27–58.
Rizo-Decelis, L., Pardo-Igúzquiza, E., Andreo, B., 2017. Spatial prediction of water quality variables along a main river channel, in presence of pollution hotspots. Sci. Total Environ. 605, 276–290.
Rizzello, S., Napoli, G., Vitolo, R., De Bartolo, S., 2024. Computer algebra software packages for master equations. https://github.com/stefano-rizzello/MasterEqSoftware.git.
Rizzello, S., Vitolo, R., Napoli, G., De Bartolo, S., 2023a. Master equation model for solute transport in river basins: part I channel fluvial scale. Stoch. Environ. Res. Risk Assess. 1–11.
Rizzello, S., Vitolo, R., Napoli, G., De Bartolo, S., 2023b. Master equation model for solute transport in river basins: part II basin fluvial scale. Stoch. Environ. Res. Risk Assess. 1–10.
Rizzoli, A.E., Young, W.J., 1997. Delivering environmental decision support systems: software tools and techniques. Environ. Model. Softw. 12 (2–3), 237–249.
Rodriguez-Iturbe, I., Muneepeerakul, R., Bertuzzo, E., Levin, S.A., Rinaldo, A., 2009. River networks as ecological corridors: A complex systems perspective for integrating hydrologic, geomorphologic, and ecologic dynamics. Water Resour. Res. 45 (1).
Rodriguez-Iturbe, I., Rinaldo, A., 2001. Fractal River Basins: Chance and Self-Organization. Cambridge University Press, Cambridge.
Shoda, M.E., Lathrop, T.R., Risch, M.R., 2015. Real-time, continuous water quality monitoring in Indiana and kentucky. U.S. Geol. Surv. Fact Sheet 3041.
Siman, K.E., Niewiarowski, P.H., 2023. Historical political ecology as qualitative social-ecological system analysis in the Maumee river watershed. Ecol. Soc. 28 (1).
Villa, F., Athanasiadis, I.N., Rizzoli, A.E., 2009. Modelling with knowledge: A review of emerging semantic approaches to environmental modelling. Environ. Model. Softw. 24 (5), 577–587.
Wing, O.E., Quinn, N., Bates, P.D., Neal, J.C., Smith, A.M., Sampson, C.C., Coxon, G., Yamazaki, D., Sutanudjaja, E.H., Alfieri, L., 2020. Toward global stochastic river flood modeling. Water Resour. Res. 56 (8).
Zaramella, M., Bottacin-Busolin, A., Tregnaghi, M., Marion, A., 2015. Exchange of pollutants between rivers and the surrounding environment: Physical processes, modelling approaches and experimental methods. Rivers–Phys. Fluvial Environ. Process. 567–590.